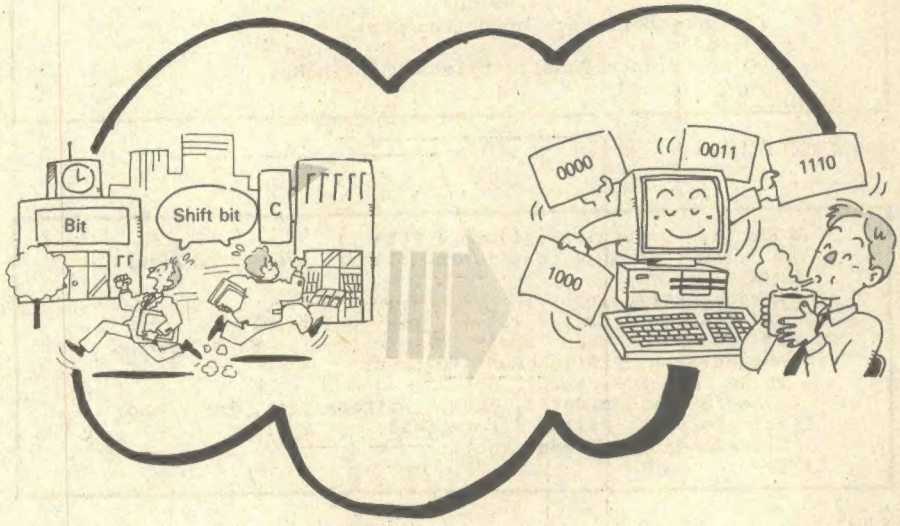
ซอฟต์แวร์
ว่าที่รต.สมหมาย แม้นมณี

SOFT MAIL

Bit ในภาษาซี : การเข้าถึงบิตข้อมูล (จบ)

ตอนจบของบิตในภาษาซี อธิบายการเข้าถึงบิตข้อมูลแบบต่าง ๆ พร้อมแสดงโฟลว์ชาร์ตการทำงานของโปรแกรม



ในตอนที่แล้วผู้เขียนได้อธิบายถึงการเข้าถึงบิตข้อมูลไปบางส่วนแล้ว พร้อมทั้งอธิบายวิธีการใช้เลขฐานสอง ในตอนนี้ซึ่งเป็นตอนจบ จะอธิบายเกี่ยวกับการเข้าถึงบิตข้อมูลในแบบต่าง ๆ พร้อมทั้งแสดงโฟลว์ชาร์ตขั้นตอนการเขียนโปรแกรมให้ดูด้วย

การ shift bit ข้อมูล

การเลื่อน (shift) บิตในที่นี้จะอธิบายถึงการ shift bit ของเลขจำนวนเต็มแบบไม่มีเครื่องหมาย โดยการ shift bit ไปทางขวาและทางซ้ายก็ได้ตามที่ต้องการ ลักษณะการ shift bit นั้นจะเป็นเลื่อนบิตไปทางขวาหรือทางซ้ายทีละบิตได้ตามต้องการ ในภาษาซีใช้เครื่องหมาย >>, << โดยที่

>> คือ การเลื่อน bit ไปทางขวาทีละ bit
<< คือ การเลื่อน bit ไปทางซ้ายทีละ bit

การ shift bit ของตัวอักขระหรือตัวเลขก็ตามเมื่อผ่านการ shift bit แล้วบิตแรกจะหายไปและเพิ่ม 0 ต่อท้าย ดังรูปที่ 1 และรูปที่ 2

```
การ shift bit ไปทางขวา (>>)
สมมติ เลขจำนวนเต็ม คือ          : 1000
      แปลงเป็นเลขฐานสองได้ดังนี้ : 0000 0011 1110 1000
กำหนดให้ตัวแปร NUM เป็นตัวแปรชนิดจำนวนเต็มแบบไม่มีเครื่องหมายเก็บเลข 1000 ไว้
      NUM = 1000;
เมื่อสั่ง  NUM = NUM >> 1;
ค่าของ NUM จะเปลี่ยนไปเป็นเลข : 500
แปลงเป็นเลขฐานสองได้ดังนี้     : 0000 0001 1111 0100
เมื่อสั่ง shift bit ต่อไปเรื่อยๆ จะได้ค่าดังตารางนี้
1000 : 0000 0011 1110 1000
 500 : 0000 0001 1111 0100
 250 : 0000 0000 1111 1010
 125 : 0000 0000 0111 1101
```

รูปที่ 1 แสดงตัวอย่างการ shift bit ไปทางขวา (>>)

```
การ shift bit ไปทางซ้าย (<<)
สมมติ เลขจำนวนเต็ม คือ          : 1000
      แปลงเป็นเลขฐานสองได้ดังนี้ : 0000 0011 1110 1000
กำหนดให้ตัวแปร NUM เป็นตัวแปรชนิดจำนวนเต็มแบบไม่มีเครื่องหมายเก็บเลข 1000 ไว้
      NUM = 1000;
เมื่อสั่ง  NUM = NUM << 1;
ค่าของ NUM จะเปลี่ยนไปเป็นเลข : 2000
แปลงเป็นเลขฐานสองได้ดังนี้     : 0000 0111 1101 0000
เมื่อสั่ง shift bit ต่อไปเรื่อยๆ จะได้ค่าดังตารางนี้
1000 : 0000 0011 1110 1000
2000 : 0000 0111 1101 0000
4000 : 0000 1111 1010 0000
8000 : 0001 1111 0100 0000
```

รูปที่ 2 แสดงตัวอย่างการ shift bit ไปทางซ้าย (<<)

จากข้างต้นสามารถเขียนเป็นแผนผัง (flowchart) และโปรแกรม ShiftBit.c ได้ดังรูปที่ 3 รูปที่ 4 และรูปที่ 5

การ rotate bit ข้อมูล

การหมุน (rotate) บิตข้อมูลจะใช้วิธีเดียวกันกับการเลื่อน (shift) บิตของเลขจำนวนเต็มแบบไม่มีเครื่องหมาย โดยการ shift bit ไปทางขวาและทางซ้ายตามที่ต้องการ แต่จะต้องมีการนำบิตที่ถูก shift ออกไป เอาไปต่อท้ายทุกครั้งที่มีการ shift bit

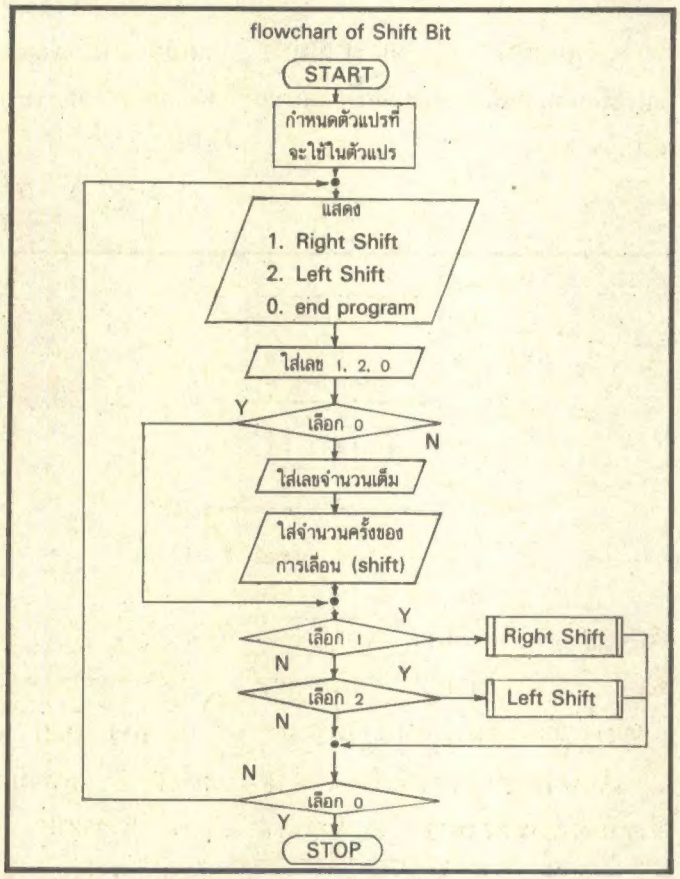
ถ้าตรวจสอบบิตที่ถูก shift ออกไปแล้วพบว่าเป็นเลข 0 ก็ทำการ shift bit ต่อไปได้ทันทีเพราะจะมีเลข 0 ต่อท้ายโดยอัตโนมัติ แต่ถ้าตรวจสอบบิตที่ถูก shift ออกไปพบว่าเป็นเลข 1 ต้องทำการเซตบิตด้านท้ายของการ shift ให้เป็นเลข 1 ทุกครั้ง ดังตัวอย่าง ต่อไปนี้

สมมติ เลขจำนวนเต็ม คือ : 13500

แปลงเป็นเลขฐานสองได้ดังนี้ :

0011 0100 1011 1100

13500_{10} = 0011 0100 1011 1100$_2$



รูปที่ 3 แสดงโฟลว์ชาร์ตการ shift bit

ถ้า rotate (>>) ไปทางขวาจะได้ดังรูปที่ 6

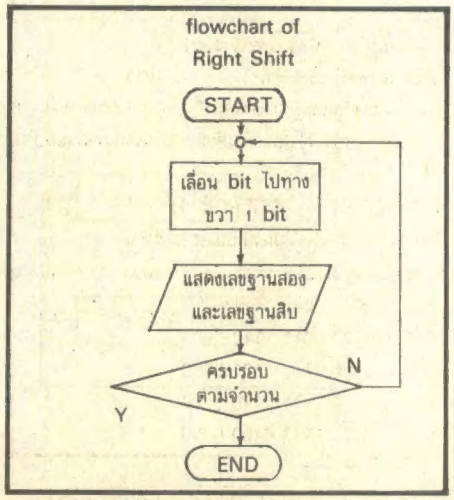
	0011	0100	1011	1100	
ครั้งที่ 1	0001	1010	0101	1110	0
ครั้งที่ 2	0000	1101	0010	1111	0
ครั้งที่ 3	1000	0110	1001	0111	1
ครั้งที่ 4	1100	0011	0100	1011	1

รูปที่ 6 แสดงการ rotate (>>) ไปทางขวา

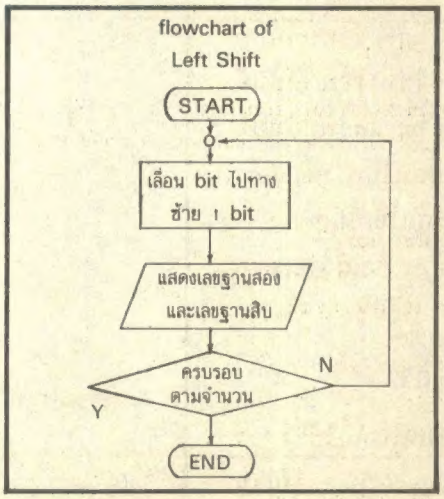
ถ้า rotate (<<) ไปทางซ้าย จะได้ดังรูปที่ 7

		0011	0100	1011	1100
ครั้งที่ 1	0	0110	1001	0111	1000
ครั้งที่ 2	0	1101	0010	1111	0000
ครั้งที่ 3	1	1010	0101	1110	0001
ครั้งที่ 4	1	0100	1011	1100	0011
ครั้งที่ 5	0	1001	0111	1000	0110

รูปที่ 7 แสดงการ rotate (<<) ไปทางซ้าย



รูปที่ 4 แสดงโฟลว์ชาร์ตการ shift bit ไปทางขวา



รูปที่ 5 แสดงโฟลว์ชาร์ตการ shift bit ไปทางซ้าย

หมายเหตุ

1. จากข้างต้นจะเห็นว่าแต่ละรอบค่าของเลขจำนวนเต็มจะเปลี่ยนไป

2. ในการเลื่อน (shift) ไปทางขวาหรือทางซ้ายแต่ละครั้งจะมีเลข 0 เกิดขึ้นที่ด้านท้ายของข้อมูลเสมอ

3. จะมีการเซตบิตด้านท้ายของการเลื่อนเป็นเลข 1 เมื่อบิตที่เลื่อนออกไปเป็นเลข 1

ขั้นตอนการหมุนไปทางขวา (Right Rotate)

1. ก่อนที่จะมีการเลื่อนไปทางขวาจะต้องทำการ check bit ด้านขวาสุดก่อนว่าเป็นเลข 0 หรือ 1

2. ถ้า check bit แล้วพบว่าเป็นเลข 0 ก็ทำการเลื่อนไปทางขวาได้ทันที

3. ถ้า check bit แล้วพบว่าเป็นเลข 1 ก็ทำการเลื่อนไปทางขวาแล้วต้องเซตบิตด้านซ้ายสุดให้เป็นเลข 1

4. ไปทำข้อที่ 1 จนกว่าจะครบรอบตามที่ต้องการ

ขั้นตอนการหมุนไปทางซ้าย (Left Rotate)

1. ก่อนที่จะมีการเลื่อนไปทางซ้ายจะต้องทำการ check bit ด้านซ้ายสุดก่อนว่าเป็นเลข 0 หรือ 1

2. ถ้า check bit แล้วพบว่าเป็นเลข 0 ก็ทำการเลื่อนไปทางซ้ายได้ทันที

3. ถ้า check bit แล้วพบว่าเป็นเลข 1 ก็ทำการเลื่อนไปทางซ้ายแล้วต้องเซตบิตด้านขวาสุดให้เป็นเลข 1

4. ไปทำข้อที่ 1 จนกว่าจะครบรอบตามที่ต้องการ

จะ check bit ได้วิธีใด

มีอยู่หลายวิธีแต่ในที่นี้ใช้วิธีการนำค่าประจำบิตไป AND กับเลขจำนวนเต็มใด ๆ ถ้าพบว่าผลลัพธ์ที่ได้เท่ากับค่าประจำบิตแสดงว่าบิตนั้นเป็นเลข 1 แต่ถ้าไม่ใช่แสดงว่าเป็นเลข 0 ตัวอย่าง

สมมติ เลขจำนวนเต็ม คือ : 13500

แปลงเป็นเลขฐานสองได้ดังนี้ : 0011 0100 1011 1100

ต้องการทราบว่า bit ที่ 3 เป็นเลข 1 หรือ 0 ทำได้ดังนี้

กำหนดให้ตัวแปร NUM เป็นตัวแปรชนิดจำนวนเต็มแบบไม่มีเครื่องหมายเก็บเลข 13500

NUM = 13500;

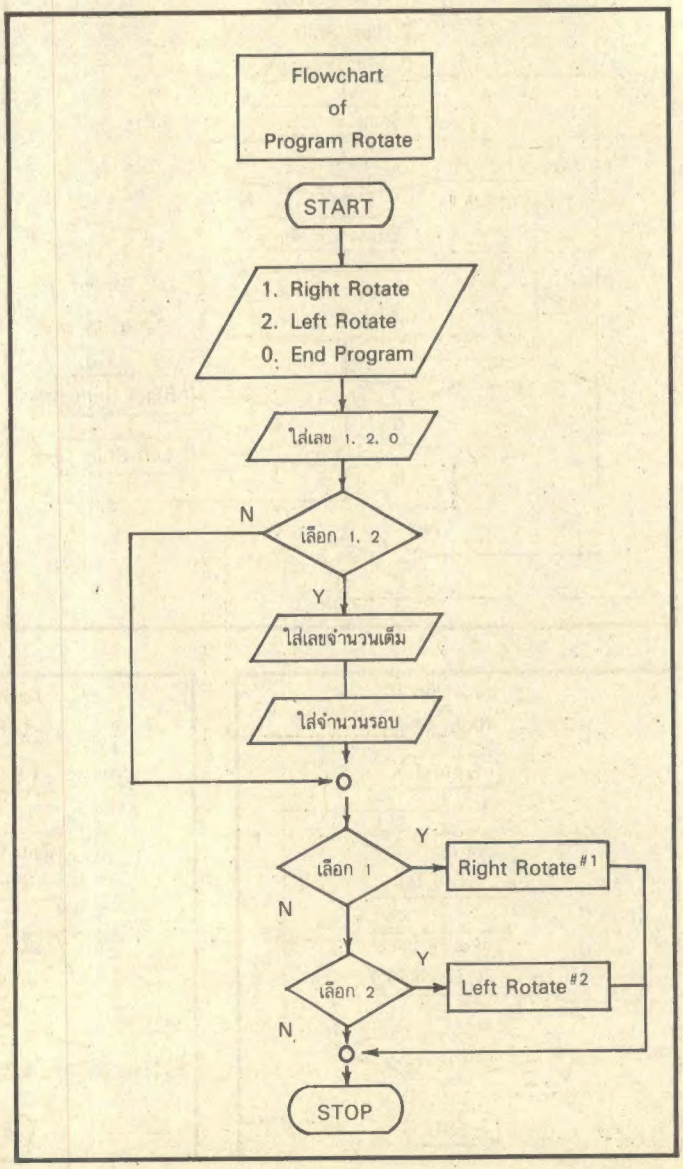
และค่าประจำ bit ที่ 3 คือ : 4

เมื่อสั่ง NUM = NUM & 4;

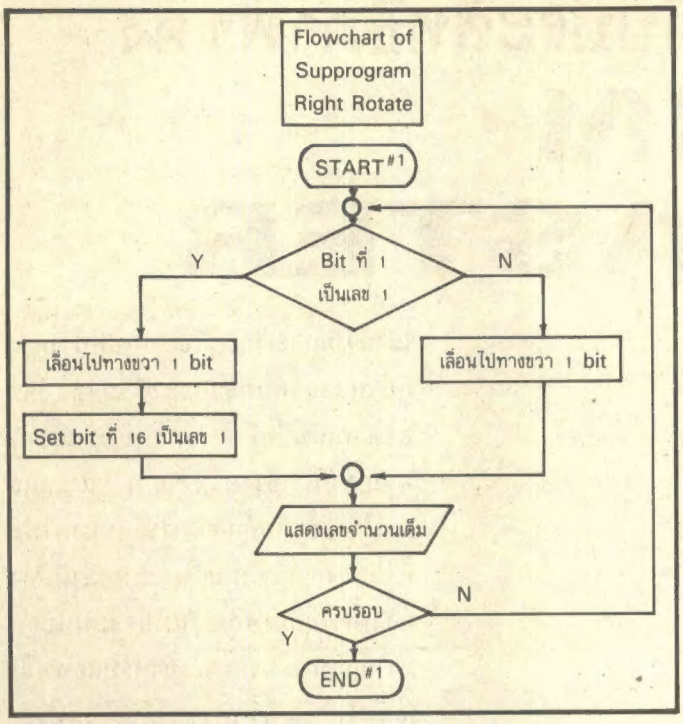
```
13500 : 0011 0100 1011 1100
&   4 : 0000 0000 0000 0100
    4 : 0000 0000 0000 0100
```

ผลลัพธ์ที่ได้เท่ากับค่าประจำบิตที่ 3 แสดงว่า bit ที่ 3 เป็นเลข 1

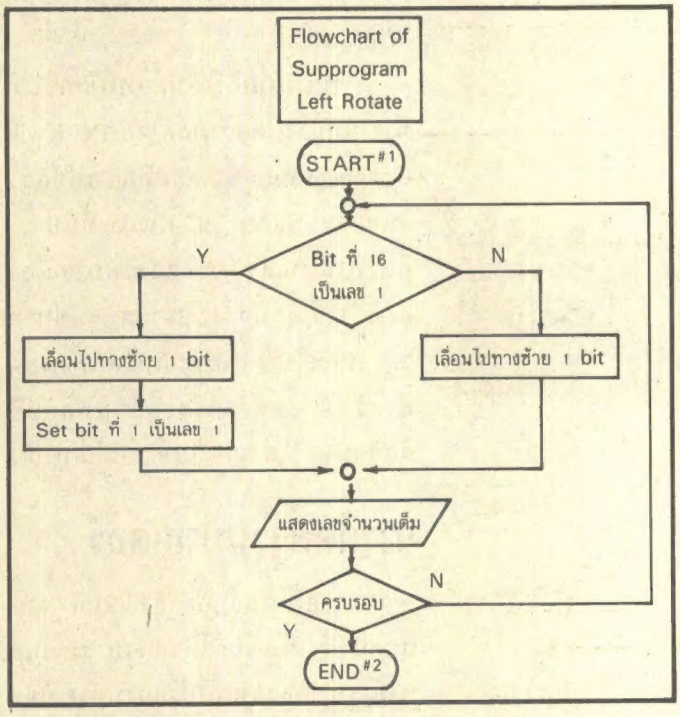
จากข้างต้นสามารถเขียนเป็นแผนผัง (flowchart) และโปรแกรม Rotate.c ได้ดังรูปที่ 8 รูปที่ 9 และรูปที่ 10



รูปที่ 8 แสดงโฟลว์ชาร์ตโปรแกรม rotate



รูปที่ 9
แสดงโฟลว์ชาร์ต
โปรแกรม rotate
ไปทางขวา



รูปที่ 10
แสดงโฟลว์ชาร์ต
โปรแกรม rotate
ไปทางซ้าย

ส่งท้าย

สุดท้ายนี้ผู้เขียนหวังว่าท่านผู้อ่านได้เข้าใจวิธีการเข้าถึงบิตข้อมูลด้วยภาษาซีตามที่เสนอเป็นอย่างดี เพราะมีตัวอย่างพร้อมกับโปรแกรมให้นำไปทดลองรันเพื่อมีความเข้าใจมากยิ่งขึ้นและสามารถนำความรู้นี้ไปประยุกต์ใช้ได้ต่อไป โปรแกรมทั้งหมดทางกองบรรณาธิการได้จัดทำเป็นซอฟต์เมล์บริการท่านผู้อ่านแล้ว สวัสดีครับ!

วางตลาดแล้ว

เทคนิคการสร้างภาพระบายเงา
และภาพเคลื่อนไหว
บน ไมโครคอมพิวเตอร์

การใช้โปรแกรม
AutoShade
AutoFlix

AutoShade และ AutoFlix เป็นซอฟต์แวร์สนับสนุนที่ใช้ร่วมกับ AutoCAD โดยที่ AutoShade ทำหน้าที่ระบายภาพ 3 มิติที่ได้จาก AutoCAD ให้มีสีสันอ่อน-แก่ หรือมืด -สว่างไล่ไปตามสภาพของแสงที่ส่องมาทำให้ได้ภาพ 3 มิติ ที่มีความเหมือนจริง ดูแล้วมีชีวิตชีวามากกว่าภาพ 3 มิติ แบบโครงข่าย (wireframe) ที่เราได้เห็นกันมาแล้วใน AutoCAD

ส่วน AutoFlix เป็นซอฟต์แวร์ที่ใช้สำหรับสร้างภาพเคลื่อนไหวซึ่งมีลักษณะคล้ายกับภาพยนตร์โดยทั่วไป คือ เป็นการนำเอาภาพนิ่งที่สว่างให้มีการเคลื่อนไหวแตกต่างกันทีละน้อยหลาย ๆ ภาพ มาฉายบนจอภาพเรียงต่อกันไปเรื่อย ๆ จนเกิดเป็นภาพเคลื่อนไหวขึ้น เหมาะสำหรับการนำเสนอโครงงานแสดงหลักการทำงานของเครื่องมือเครื่องจักรหรือกระบวนการทำงานต่าง ๆ ตลอดจนใช้สร้างสื่อการเรียนการสอนได้เป็นอย่างดีอีกด้วย

นอกจากนี้ AutoFlix ยังใช้กับเครื่อง PC โดยทั่วไปที่ไม่มี math co-processor ได้เช่นกัน ดังนั้นนักเล่นคอมพิวเตอร์กราฟฟิกจึงไม่ควรพลาด

หาซื้อได้ตามร้านหนังสือชั้นนำทั่วไป

ซีเอ็ด